

Application No. 09/742,720  
Response to Final Rejection dated November 18, 2005  
Reply to Final Office Action dated September 19, 2005  
Express Mail EV723448241US

Remarks

The Office Action dated September 19, 2005 has been noted, and its contents carefully studied. In light of the following discussion, reconsideration of the rejection of the claims under 35 U.S.C. §103 is courteously requested.

Initially, it is noted that this Office Action is a Final Rejection based on a newly cited reference which the Examiner has asserted was necessitated by Applicants' substantial amendment, i.e., by incorporating new limitations into the independent claims, which allegedly require further search and consideration. Applicants respectfully disagree with the Examiner's characterization of the amendments. More specifically, as to claim 1, all that Applicants have previously done is to recharacterize the first step by stating that instead of determining if the client is on the same system as the server, requiring a query by the server to determine if the client is on the same system as the server. In this regard, this is substantially the same issue and limitation as further clarified for the first step so that no new search would be required. The change to the second clause merely changes the terms "establishing an interprocess communications facility connection" to "indicating that if the query indicates the client is on the same system as the server as a local client, setting pointers to establish the interprocess communications facility". The recitation of the interprocess communications facility as being a transport layer interfaced (TLI) was previously added from original claim 5, and that clause has now merely been clarified as being to bypass the connection oriented protocol. The third clause has been deleted as being merely redundant to the modifications to the second clause, and the amendment to the third clause merely clarifies that data is transferred directly between the client and the server "within the same connection as the connection oriented protocol".

With respect to the amendments to claim 9, they are substantially of the same nature as those to claim 1. As such, it is respectfully urged that Applicants' amendments did not necessitate a new ground of rejection and further search. The Examiner has conducted such a search of his own accord and it is respectfully urged that the finality of this Rejection should be withdrawn and full consideration be given to Applicants' arguments advanced herein before any Final Rejection is issued.

Application No. 09/742,720  
Response to Final Rejection dated November 18, 2005  
Reply to Final Office Action dated September 19, 2005  
Express Mail EV723448241US

With respect to the current amendments to the specification and to claim 1, they are to correct minor grammatical errors and should be entered. The specification amendment is to change "connection" to "connect". The claim amendment change adds "is" after "client" to make the clause grammatically correct.

It is also noted that this Response is being filed within two months of the date of the Final Rejection for purposes of calculating any extension of time periods. It is noted however, that such policy regarding extensions of time periods is not appropriate in the instant case because as previously discussed, the Office Action should not have been a Final Rejection.

Turning now to the invention as recited in the claims, it is respectfully urged that while the cited reference provides a method and system for executing self calls when a client and a server are one and the same, the method and system described therein are substantially different and unrelated to the method and system claimed by Applicants.

More specifically, in one aspect Applicants provide a method of interprocess communications between the client and a server. Each client and server has one or more interprocess communications facilities which are sockets, and each interprocess communications facility has connection oriented protocol (COP) associated therewith. The method provides for initiation of a query by the server to determine if the client is on the same system as the server. In this regard, what is meant by the term "local" or "on the same system" means that the client and server are both located on the same machine and are not connected by a network connection such that the connection oriented protocol is not necessary for the accurate transfer of data (see paragraph 4 of the specification).

If the query indicates the client is on the same system as the server as a local client, pointers are set to establish an interprocess communications facility connection between the server and the client. The interprocess communication facility is specifically required to be a Transport Layer Interface (TLI) to bypass the connection oriented protocol. Data is then transferred directly between the client and the server within the same connection as the connection oriented protocol in a manner bypassing the connection oriented protocol. The transferring of the data further includes detecting any errors in the data transferring step such that

Application No. 09/742,720  
Response to Final Rejection dated November 18, 2005  
Reply to Final Office Action dated September 19, 2005  
Express Mail EV723448241US

if errors are detected, pointers are set to null and data is transferred through the connection oriented protocol.

Claim 9 provides the invention in a system aspect specifying the configuration of the server in a manner for conducting the method previously described, and in which the interprocess communications facility is clearly defined as being a Transport Layer Interface, with the communications being conducted through sockets connected through the Transport Layer Interface (TLI), (reference is made to paragraph 32 which discusses the transport layer interface (TLI) connections as being between sockets 35 and 37 as described in the specification).

Claim 7 provides the step of determining if the server Interprocess Communications Facility, i.e., the TLI for the server, and the client Interprocess Communications facility, i.e., the TLI for the client, within the same facility are compatible so that the COP can be avoided. If the facilities are not compatible, then the data is transferred between the client and the server through a conventional COP connection. These features are also recited as independent system claim 9. Thus, as further recited in other dependent claims, the invention contemplates setting of pointers between respective sockets within the same system containing both the client and the server.

It is respectfully urged that the invention as recited in the claims is not obvious under 35 U.S.C. §103 from the newly cited reference, as will become more clearly evident from the following detailed discussion of the reference, presented herein for the Examiner's kind consideration.

U.S. Patent No. 6,728,788 to Ainsworth et al.

U.S. Patent No. 6,728,788 to Ainsworth et al. (hereinafter "Ainsworth") discloses a data processing system and method for interprocess or intraprocess communication, specifically remote procedure calling (column 1, lines 10-13). The Examiner has asserted that Ainsworth discloses a server having server data and a server Interprocess Communications Facility which is a socket associated therewith, with the server being configured for communicating with one or more clients having client data and a client Interprocess Communications Facility which is a socket associated therewith. In so stating the grounds for the Rejection, the Examiner has cited Figure 5C.

Application No. 09/742,720  
Response to Final Rejection dated November 18, 2005  
Reply to Final Office Action dated September 19, 2005  
Express Mail EV723448241US

Applicants respectfully disagree with the Examiner's characterization of the disclosure of Ainsworth. All that Figure 5C shows is a method at client run time in which a client process executes code that has been previously generated using a process similar to that depicted in Figure 5B (column 9, lines 20-22). At run time, client code determines from a binding if the current RPC call is directed to the same process in which the client code is running (step 512). If the call is directed to the same process, the client code then performs a lookup in the interface registry to get the local address of the manager function (step 514). The client code then converts the binding from a client-side binding to a server-side binding (step 516), and the local procedure call is made (step 518) (column 9, lines 23-31).

Specifically, Ainsworth clearly states that in order to implement the self-call mechanism of the invention within the DCE, three changes need to be made to an IDL compiler to support self call. More specifically, the first change provides a new, user-specifiable option which indicates whether self call codes should be generated. The user may specify an option that indicates whether self call codes should be generated. This is done by a new command line option flag, an option in a configuration file, etc which is accepted by the compiler which stores the user preference in its option table along with other user specifiable options. The second modification is the addition of a function of the code generator of the IDL compiler to generate the code to check whether self call is supported. The third modification is the addition of a function to the code generator of the IDL compiler to generate the code to call the manger interface (see generally column 10, lines 41-64).

While these are specific details of how self call is generated, which is not even remotely related to Applicants' invention, a more general implementation description of self call is set forth in column 9. As generally described therein, self call invokes a separate IDL compilation for switching calls from RPC to local calls. At run time, the client code determines from the binding if the current RPC call is directed to the same process in which the client code is running. If so, the client code then performs a lookup in an interface registry to get the local address of the manager function. The client code then converts the binding from a client-side binding to a server-side binding, and the local procedure call is made (see generally column 9, lines 1-31).

Application No. 09/742,720  
Response to Final Rejection dated November 18, 2005  
Reply to Final Office Action dated September 19, 2005  
Express Mail EV723448241US

In distinguishing over the reference, it is important to appreciate that Applicants' claims clearly state that the query by the server determines if the client is on the same system as the server. If so, pointers are set to establish an Interprocess Communications Facility connection between the server and the client, which as clearly described and specified in the application, relates to pointers to sockets. Applicants' claims further call for the Interprocess Communication Facility as being a transport layer interface (TLI) to bypass the connection oriented protocol. This is clearly not taught or suggested by Ainsworth which requires a complex interaction with a compiler to generate three types of code to enable the local connection.

The distinctions between Applicants' inventions and the teachings of Ainsworth are further supported by the disclosure of Ainsworth itself under the description of related art. More specifically, Ainsworth clearly states that by convention, communication architectures in a local area network are typically characterized as conforming to a seven layer model in the following hierarchy: physical layer, logical link layer, network layer, transport layer, session layer, presentation layer, and application layer (column 1, lines 23-27). Ainsworth further states that the session layer typically provides remote procedure call (RPC) support, maintains the integrity of the connection between nodes and controls data exchange (column 1, lines 36-39). Ainsworth clearly processes its calls at the session layer and then provides a complex compiler generation of code to achieve the self call.

Applicants' invention again, it is respectfully pointed out, requires that the pointers establish the interprocess communications facility connection as a transport layer interface (TLI). Ainsworth in contrast, describes its self call technique as being an RPC optimization scheme called "self call" that will completely bypass the RPC parameter, marshalling/unmarshalling and the transport stack altogether for those RPC's that qualify. Thus, in no remote stretch of the imagination can Ainsworth be considered as establishing a self call through a Transport Layer Interface because it clearly states that it bypasses the transport stack (column 7, lines 15-33).

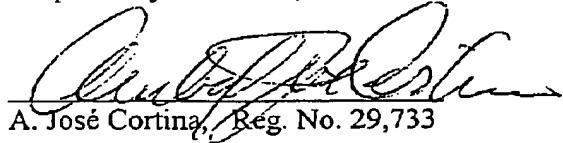
Thus, for the foregoing reasons, it is respectfully urged that the claims clearly define patentable subject matter under 35 U.S.C. §103. It is also again respectfully urged that for the reasons set forth previously that the finality of the Rejection was inappropriate and should be withdrawn, and that the Examiner should fully consider Applicants' comments herein.

Application No. 09/742,720  
Response to Final Rejection dated November 18, 2005  
Reply to Final Office Action dated September 19, 2005  
Express Mail EV723448241US

Nonetheless, should the Examiner have any questions, comments or suggestions of a nature necessary to clarify the claims and to place the case in condition for allowance, he is courteously requested to telephone the undersigned at the number listed below.

Dated: November 18, 2005

Respectfully submitted,



A. José Cortina, Reg. No. 29,733

Daniels Daniels & Verdonik, P.A.  
P.O. Drawer 12218  
Research Triangle Park, NC 27709  
Voice 919.544.5444  
Fax 919.544.5920  
Email jcortina@d2vlaw.com

Enclosures

F:\CL\1301-006\Prosecution\Response Final Rejection.Nov.2005.doc